Esecurity: secure internet & e-passports, summer 2011

MICHAEL NÜSKEN, RAOUL BLANKERTZ

7. Exercise sheet
Hand in solutions until Sunday, 22 May 2011, 23:59

Exercise 7.1 (Project). (12+12 points)

Choose whether you consider SSL or SSH for this exercise. 12+12

Find sources that describe the chosen protocol and study them. These sources should include the relevant up-to-date RFCs. Supply a list of all used sources! Give a short description of the protocol (in your own words!), enough to answer the following security questions.

Discuss the security of the chosen protocol under the same security aspects as we did for IPsec:

(i) Session key agreement.

(ii) Perfect forward security.

(iii) Denial of Service.

(iv) Endpoint identifier hiding.

(v) Live partner reassurance.

(vi) Plausible deniability.

(vii) Stream Protection.

(viii) Negotiating parameters.

We will summarize your results in the course and tutorial on 24 May.

Exercise 7.2 (AtE and died: confidentially poisoned). (10+2 points)

Horton's principle says that one should always prove the integrity of the *message text*. One solution to ensure the integrity is to first authenticate and then encrypt (AtE). Though this paradigm is clearly correct and the conclusion grants integrity as desired, we overlooked a different issue here. This exercise shall prove it.

Suppose we use some encryption function ENC_{K_e} and any message authentication function MAC_{K_a}. For a message m we compute $a := \mathrm{MAC}_{K_a}(m)$ and send $c := \mathrm{ENC}_{K_e}(m|a)$. (Here, the vertical line '|' denotes concatenation.)

Assume both are as secure as you like. In particular, the encryption function shall guarantee that even to a *c*hosen *m*essagetext *a*ttacker the encryptions of two known plaintexts are *ind*istinguishable. In other words, there is no (ie. no probabilistic polynomial time) so-called IND-CPA attacker: the attacker may ask for encryptions of chosen plaintexts and he fixes two further message texts m_0, m_1 for which he never inquired the encryption. Finally, the attacker is given the encryption of m_0 or of m_1 and shall tell which of the two message texts was used. One possible encryption function under these constraints is the one-time pad (assuming that the encryption procedure keeps track of the already used parts of the key).

Now, suppose additionally that the encryption XORs something on the plaintext (like a one-time-pad), and define a variant $\mathrm{ENC}^*_{K_e}$ of this encryption function as follows: first replace every 0-bit by two bits 00 and every 1-bit by two bits 01 or 10, choose randomly each time, next encrypt with ENC_{K_e}. For the decryption we translate 00 back to 0, 01 and 10 to 1, and 11 is considered as a transmission error. So we send $\mathrm{ENC}^*_{K_e}(m|\,\mathrm{MAC}_{K_a}(m))$.

2+2 (i) Prove (at least, argue) that $\mathrm{ENC}^*_{K_e}$ is still secure in the previous sense.

4 (ii) Suppose that a ruthless person, called Raoul, has overheard the messages of your login to some server which was done by sending the password. Of course, your password was authenticated and encrypted, as all messages. Now, Raoul takes the transmission of your password and resends it with a bit pair in the cipher text inverted.

  (a) How does the recipient react if the original bit was 0?
  (b) How does the recipient react if the original bit was 1?

  Conclude that Raoul learns the bit from the reaction of the server (and thus your passwords after enough trials).

2 (iii) Estimate the effect of this observation.

2 (iv) In SSH we transmit $\mathrm{ENC}_{K_e}(m)|\,\mathrm{MAC}_{K_a}(m)$, so we authenticate and encrypt (rather than first authenticating and second encrypting). Is that better? [Try to use $\mathrm{ENC}^*_{K_e}$ here.]